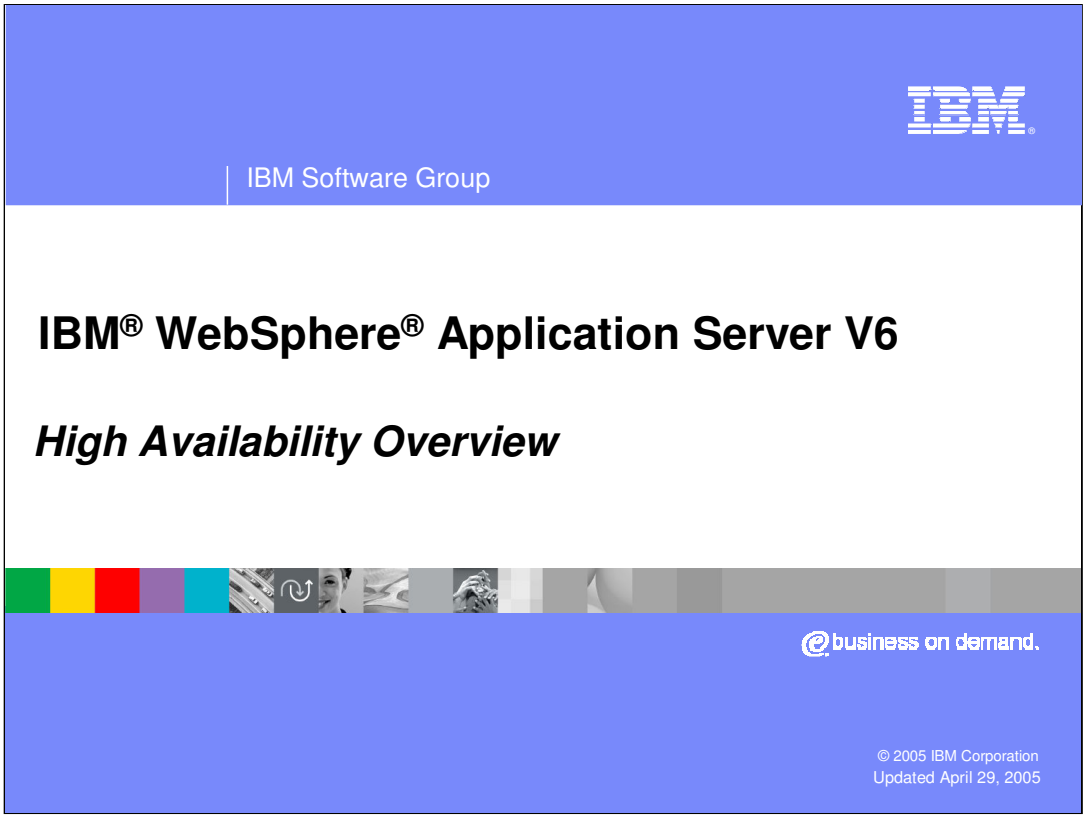


This presentation will cover the High Availability services in WebSphere Application Server V6 at a basic level.

Goals

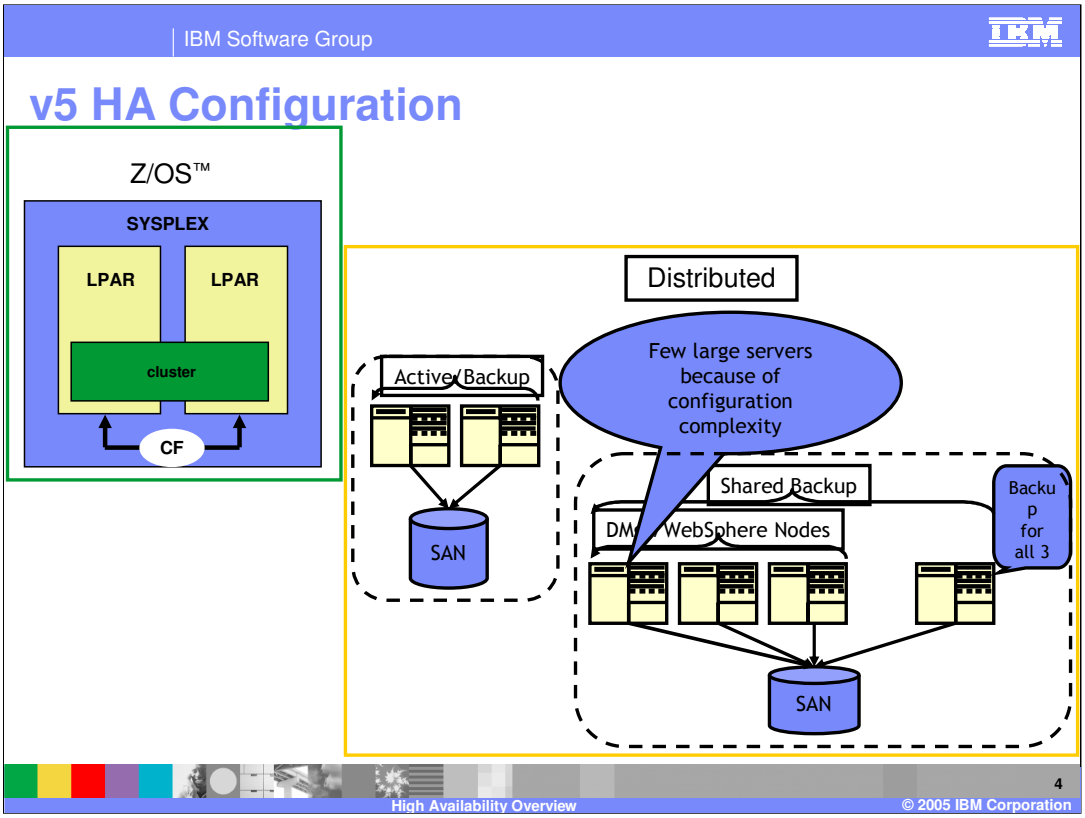
- Describe the benefits of High Availability services
- Compare capabilities between V6 and V5

The goals of this presentation are to describe WebSphere Application Server's new High Availability services and to compare these new capabilities to the capabilities of WebSphere Application Server V5 in this area. More details on this topic, including how to configure High Availability services, can be found in the presentation titled "High Availability Details including Core Groups and Policies".

WebSphere Application Server V5: “HA” Status

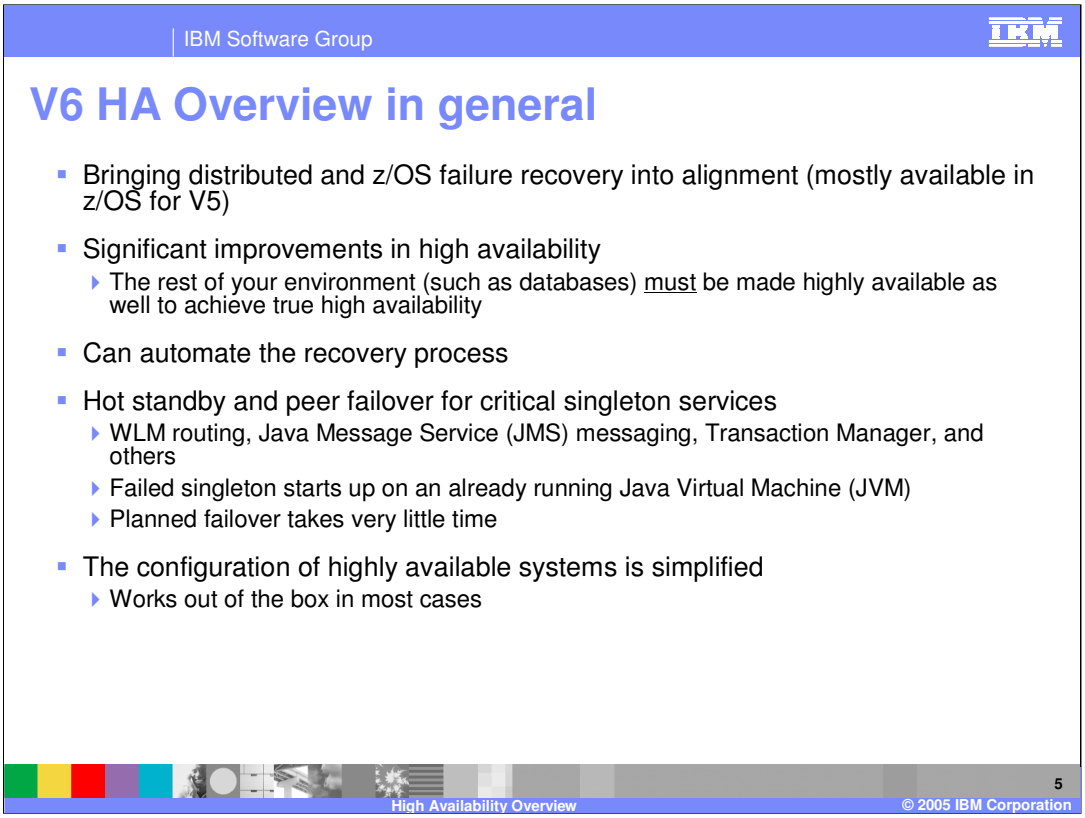
- Cold failover only
  - Peer Restart and Recovery (PRR)
    - Clears in-flight transactions
    - Releases locks
  - Normal start of backup server
  - Still available in V6 as an option
    - Supports V5 workloads

With WebSphere Application Server V5, there are some obstacles to achieving true high availability. In version 5, High Availability meant Peer Restart and Recovery automated with z/OS ARM. Failover lead to several minutes of downtime, which is not ideal.



The picture for distributed platforms that required high availability tend to be configured in either Active/Backup pairs, or with a few active nodes and a single machine acting as backup for all of them. Because of the complexity of this configuration, it was easier for most people to use only a few machines, which meant they had to be quite large if they were handling a heavy workload.

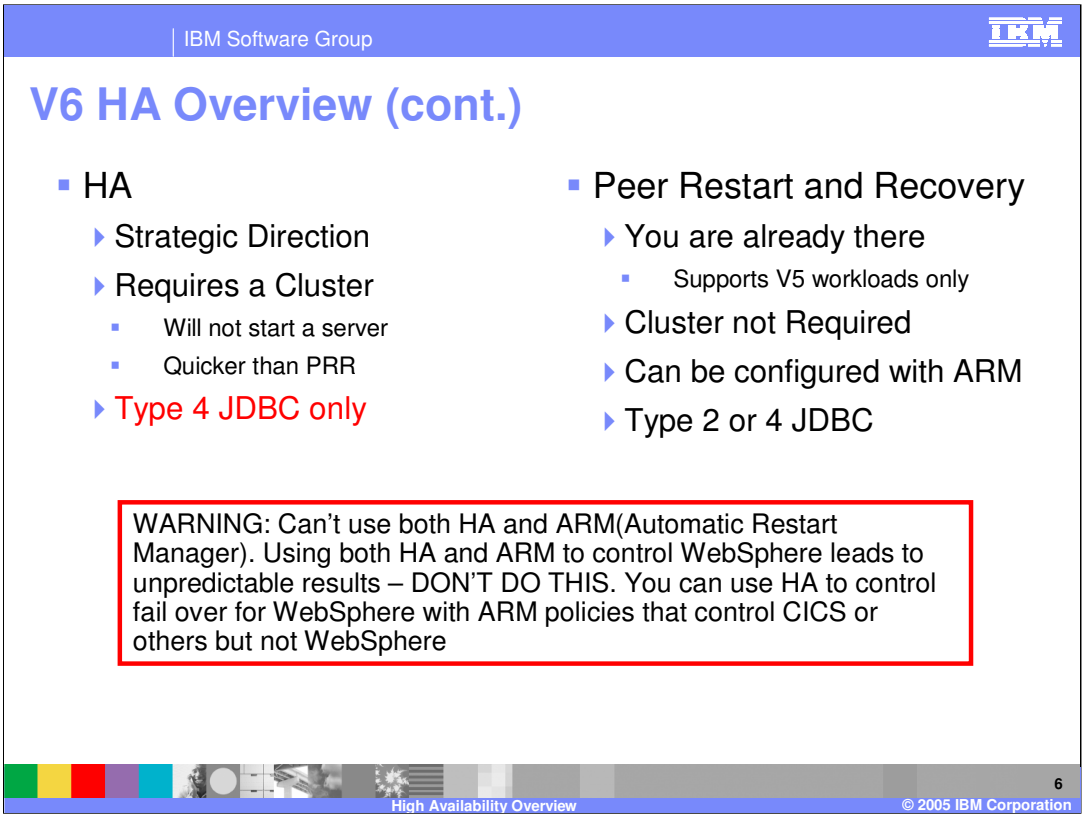
This picture was different for z/OS. Here RSS and ARM played key roles in Peer Restart and Recovery, and still can.



WebSphere Application Server Version 6 brings the distributed and z/OS platforms into closer alignment. WebSphere Application Server V6 can be used as part of a highly availability environment as a result of a new service known as the High Availability Manager. The High Availability Manager runs important services on any server that is available. The High Availability Manager keeps track of the status of all of your servers and the services that they are running, ensuring that all services remain continuously available. When a failure is detected, the failed service can be started in another already-running JVM, potentially on another physical machine, in very little time. Planned failover takes less than a second.

While many of these properties were true for PRR, HA and PRR are very different.

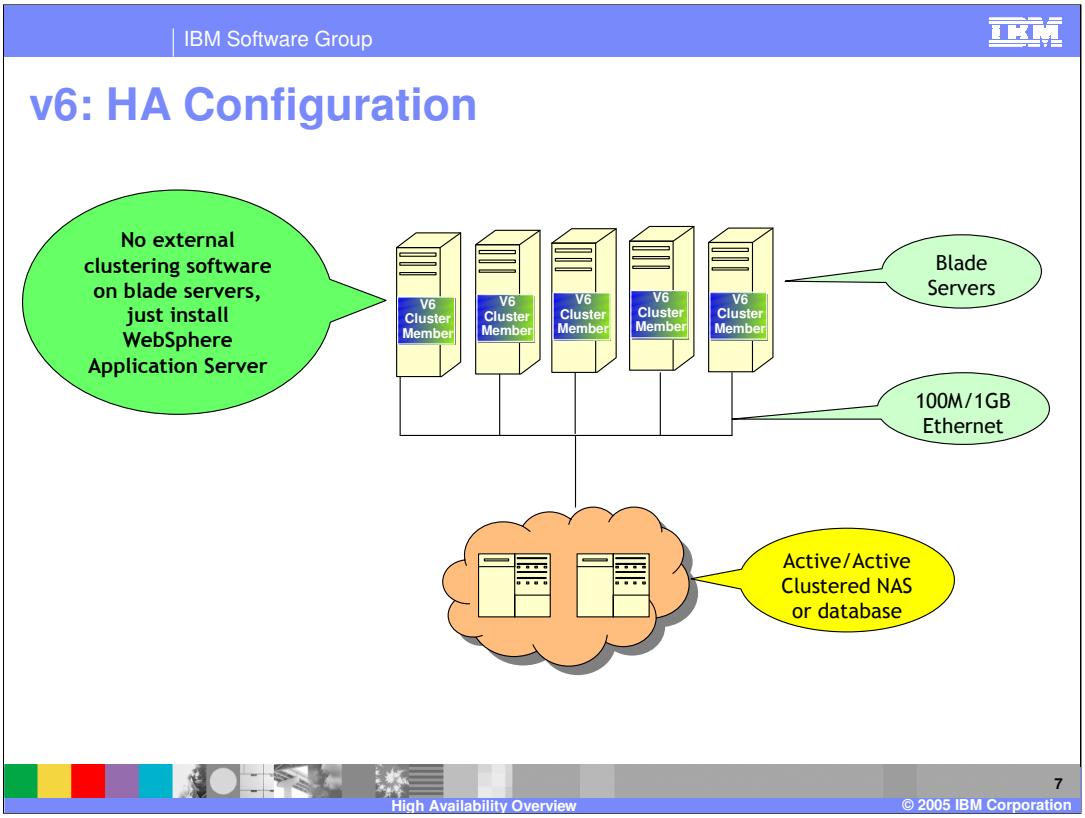
This new capability is far easier to configure than PRR. In most cases, this capability will work out of the box, with no configuration required.



The best practices is to start using WebSphere Application Server V6 with PRR and move to HA over time. HA is the strategic direction for WebSphere, but HA and PRR are different. You understand PRR, and need to learn and move to HA, in a controlled fashion. PRR development will not advance much farther and will not work with

1) Web Services
2) JCA 1.5 connectors
3) System Integration Technologies
4) and many future J2EE paradigms

The type 4 JDBC is an RSS restriction and a tech doc describing this restriction is included with the GA delivery of WebSphere Application Server V6.

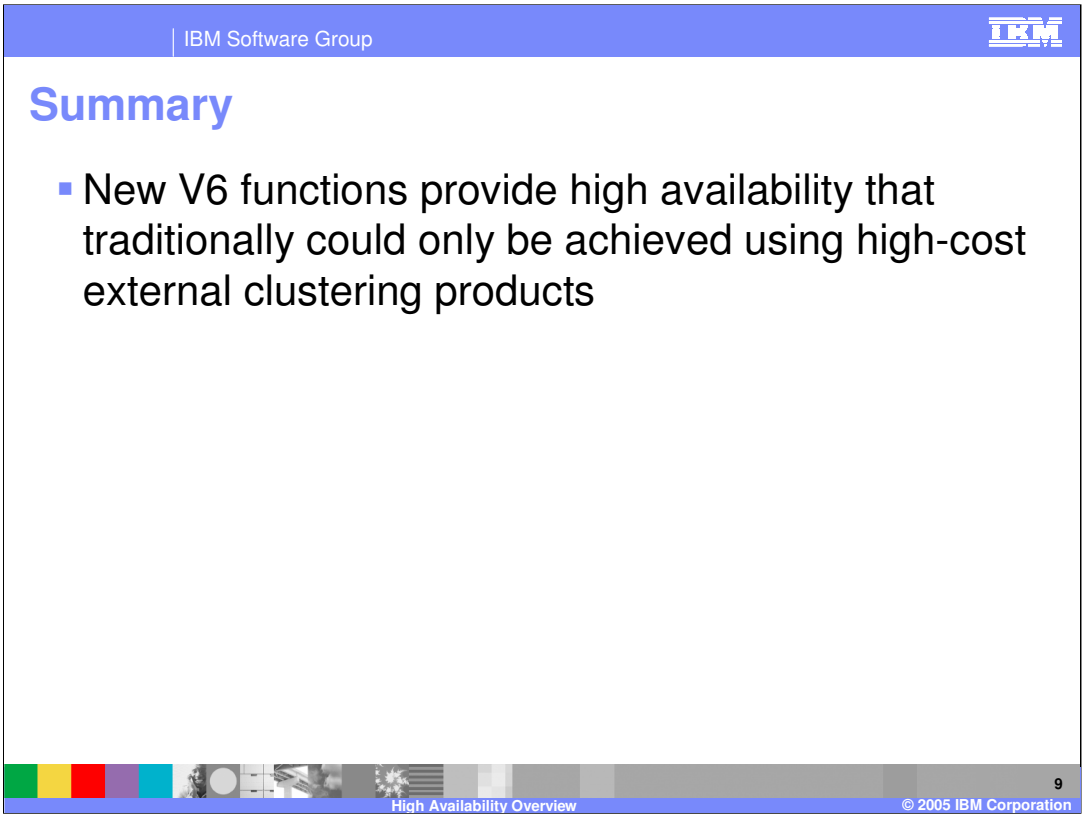


It is far easier to build a highly available environment with WebSphere Application Server V6. You need only to install WebSphere Application Server on your servers. Because no external clustering software is required, it is more practical to deploy a large network of blade servers. However, you will need to ensure that any resources on which WebSphere Application Server depends are made highly available. This means storing transaction logs on a highly available Network Attached Storage system or logstreams, and making sure that your database servers are clustered, among other things. The real net gain is unified management and quicker transition times.

Failover for Different Service Types

- Singleton services that require failover fall into two categories:
  - User resources: application related resources
    - Example: Transaction log for 2PC transaction, Messaging Engine
    - Failover occurs only within the cluster boundary
  - System resources: used internally by WebSphere Application Server
    - Example: WLM routing
    - Failover can occur to any process within the Core Group
- A Core Group defines a set of processes that can cooperate to provide each other with high availability

Singleton services that require failover can be divided into two categories, user resources and system resources. User resources are artifacts related to a user application, such as a transaction log or a messaging engine. These resources can fail over only to another member of the same cluster, because only members of the same cluster are guaranteed to be running the same applications. System resources, such as WLM routing, can be run on any process within the same Core Group. A Core Group is a boundary that defines the set of processes that can provide each other with high availability. In the default and most common configuration, all processes are members of a single Core Group.



In summary, this presentation has focused on the new high availability related functionality in WebSphere Application Server V6. This functionality provides far better availability than WebSphere Application Server V5 and does not require the assistance of external clustering products. WebSphere Application Server V6 is capable of being part of a carefully designed environment that provides high availability. More details on this topic, including how to configure High Availability services, can be found in the presentation titled “High Availability Details including Core Groups and Policies”.

Trademarks, Copyrights, and Disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	CICS	IMS	MQSeries	Tivoli
IBM(logo)	Cloudscape	Informix	OS/390	WebSphere
e(logo)business	DB2	iSeries	OS/400	xSeries
AIX	DB2 Universal Database	Lotus	pSeries	zSeries

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.